The method is for constructing predictive models that can be used to make predictions in situations where the inputs to those models can have values that are missing or are otherwise unknown.

That is, the method includes presenting a collection of training data comprising examples of input values that are available to the model together with corresponding desired output value(s) that the model is intended to predict, and generating a plurality of subordinate models, that together comprise an overall model. Each subordinate model has an associated set of application conditions that must be satisfied in order to apply the subordinate model when making predictions.

The application conditions include tests for missing values for all, some, or none of the inputs, and tests on the values of all, some, or none of the inputs that are applicable when the values of the inputs mentioned in the tests have known values.

For at least one subordinate model, the training cases used in the construction of that subordinate model include some cases that indirectly satisfy the application conditions such that the application conditions are satisfied only after replacing one or more known data values in these training cases with missing values.

Further, as exemplarily defined by independent claim 1, the method further includes "*outputting a specification of at least one of said subordinate models thus generated and making a prediction based on said at least one of said subordinate models thus-generated.*" In one embodiment, the specification of a plurality of subordinate models and their associated application conditions, are output to a storage device for being read by the machine, thereby enabling the plurality of models to be readily applied to generate predictions.

With the unique and unobvious features of the claimed invention, the method can realize significant advantages because it can be readily applied in conjunction with any known method for constructing models, including ones that require all input values to be known, thereby yielding combined methods for constructing models that tolerate missing values.

In an exemplary embodiment, the method and storage device storing the method, can be utilized in combination with classification and regression trees, classification and regression rules, or stepwise regression (e.g., see page 9 of the present specification).

II. THE §101 Rejection

Claims 1-20 stand rejected under 35 U.S.C. §101. Specifically, the Examiner asserts that:

> *2.1 Product claims 1-20 are rejected because the underlying process invention comprises an abstract idea.*
>
> *2.2 Regarding Claim 1, this claim is directed to "A program storage device readable by a machine, tangibly embodying a program of instructions executable by the machine to perform method steps for constructing predictive models", and the steps recited in Claim 1 describe mathematical operations comprising the abstract idea of generating models that account for missing or otherwise unknown data values.*
>
> *For the purposes of examination, the "device" of claims 1-20 will be read broadly to comprise a product claim that encompasses any and every computer implementation of a process. Neither the detailed description of the invention nor the drawings supply any tangible description of a computer implementation of the invention.*
>
> *In this situation, the following paragraph in the Guidelines at IV.B.2.(a)(ii) appears controlling:*
>
> *If a claim is found to encompass any and every product embodiment of the underlying process, and if the underlying process is statutory, the product claim should be classified as a statutory product. By the same token, <u>if the underlying Process invention is found to be non-statutory. Office personnel should classify the "product" claim as a "non-statutory product."</u> If the product claim is classified as being a non-statutory product on the basis of the underlying process, Office personnel should emphasize that they have considered all claim limitations and are basing their finding on the analysis of the underlying process.*
>
> *[Emphasis supplied.]*
>
> *Therefore, Claim 1 is rejected as being classified as a non-statutory product because the underlying process invention as claimed by Applicant is non-statutory. The method steps in Claim 1 do not: (1) recite data gathering limitations or post-mathematical operations that might independently limit the claims beyond the performance of a mathematical operation; or (2) limit the use of the output to a practical application providing a useful, concrete, and tangible result.*
>
> *2.3 Regarding Claims 2-20, the limitations supplied in these claims do not: (1) recite data gathering limitations or post-mathematical operations that might independently limit the claims beyond the performance of a mathematical operation; or (2) limit the use of the output to a practical*

> *application providing a useful, concrete, and tangible result. The analysis and conclusion regarding non-statutory subject matter is identical to Claim 1 above.*

Applicant respectfully disagrees for all of the reasons submitted in the March 17, 2000, Amendment incorporated herein by reference.

Additionally, Applicant has provided still further reasons clearly establishing the statutory nature of the invention. While the Examiner presumably believes that the claims must recite the <u>exemplary application of the invention to the direct-mail marketing</u>, Applicant submits that such an amendment to the claims is unnecessary and indeed would serve to unduly limit the invention for no apparent reason or purpose.

More specifically, the Office Action erroneously asserts that applicant has disclosed methods and apparatuses for using a computer but no practical application is discussed in the specification or in the claims. Furthermore, the Examiner asserts that the specification and claims merely discuss performing the abstract idea of generating models that account for missing or otherwise unknown data values. The Office Action also incorrectly asserts that no practical application of the invention is discussed, that none of the embodiments performs any post-computational processing activities, and that data is not extracted from a mathematical calculation to be manipulated to achieve a practical activity. Applicant respectfully submits that the Examiner's rejection is erroneous.

a. Computer Implementations

First, regarding the Examiner's assertion that "*[n]either the detailed description of the invention nor the drawings supply any tangible description of a computer implementation of the invention*" (Section 2.2 of the Final Office Action, second paragraph, second sentence), Applicant respectfully disagrees.

As stated on Page 19, Lines 21-24, of the detailed description of the invention, "*[t]he steps are presented in such a way that they may be readily combined with any method for constructing the subordinate models of the plurality, including ones that require all input values to be known.*"

In particular, the preferred method steps describe how to combine the invention with

stepwise regression, classification and regression trees, and classification and regression rules. A deliberate effort was made to phrase the method steps in such a way that someone ordinarily skilled in the art of implementing any one of the aforementioned predictive modeling techniques could, upon reading the disclosure and the cited literature, implement the invention for that technique.

Moreover, the method steps were phrased in a way that anticipates the possibility of combining aspects of all three types of predictive modeling techniques in a single algorithm. The purpose was to enable someone ordinarily skilled in the art to apply the invention in a much broader context than is implied by any one of the afore-mentioned predictive modeling techniques.

Method Step 1 is present in all three of the aforementioned predictive modeling techniques; each begins with some initial model that is then refined.

Method Steps 2a and 2b address stepwise regression. This predictive modeling technique repeatedly performs incremental model refinement steps on an initial regression equation until a set of stopping conditions are met (e.g., "*until it is decided that no further refinements are justified*").

The refinement steps comprise adding variables (e.g., input data fields) to, or removing variables from, a current regression equation to produce a new regression equation. The new regression equation then becomes the current regression equation, thereby enabling further refinements to be performed. The various ways of implementing stepwise regression are well-known to those ordinarily skilled in the art of programming stepwise regression algorithms.

Stepwise regression does not consider a plurality of models, but instead repeatedly refines a single model. The detailed description of the invention teaches the advantages of maintaining a plurality of models using a regression problem as an example. Computer methods for maintaining a plurality of regression equations should be (and indeed are!) self-evident to one ordinarily skilled in the art of computer programming.

For the sake of argument, maintaining and utilizing associated application conditions, on the other hand, might not be self-evident unless one is also knowledgeable about classification and regression tree algorithms and/or classification and regression rule algorithms. The cited literature on these topics teaches computer methods for implementing

the application conditions required by the invention. Armed with this knowledge, those ordinarily skilled in the art of programming stepwise regression algorithms would then be able to implement Method Steps 2a and 2b of the invention.

Method Step 2c addresses classification and regression trees. These predictive modeling algorithms already construct pluralities of models. In this case, a plurality comprises the models at the leaves of a tree, and the application condition of each such model is the conjunction of the branch conditions along the path leading from the root of the tree to the corresponding leaf. Classification and regression tree algorithms repeatedly perform incremental model refinement steps on an initial tree (usually a single root node) until a set of stopping conditions are met (e.g., "*until it is decided that no further refinements are justified*"). Each refinement step comprises adding two or more child nodes to a leaf node in the current tree. The child nodes are assigned disjoint branch conditions and they then become new leaf nodes. The method of constructing tree branches is thus directly analogous to that described in Method Step 2c. The various ways of implementing such refinement steps are well-known to those ordinarily skilled in the art of programming classification and regression tree algorithms.

Method Step 2c specifies the preferred method of modifying classification and regression tree algorithms to incorporate the invention by specifying the preferred method for constructing tree branches using the invention.

Some classification and regression tree algorithms treat a "missing" value as a legitimate data value. Tests for missing values thus appear in various branch conditions. These algorithms exemplarily employ a version of the prior art method discussed in the Summary of the Invention beginning on Page 3, Line 15: "METHODS THAT INTRODUCE 'MISSING' AS A LEGITIMATE DATA VALUE".

To incorporate the invention in these algorithms, the same prior art methods for constructing tree branches would be used. However, as discussed in the Detailed Description of the Invention, the training cases used to construct the models that appear in each tree node would preferably be those that indirectly satisfy the application conditions of the model for those missing values that are mentioned in the application conditions and that are to be treated as missing at random.

The latter distinction is a fundamental difference between the invention and the prior

art classification and regression tree methods; hence, the difference forms a basis for the patentability of the combination of the patent claims.

As discussed in the Detailed Description of the Invention, if none of the missing values mentioned in the application conditions is to be treated as missing at random, then only those training cases that directly satisfy the application conditions would preferably be used to construct the associated subordinate model, as per the prior art method. It is this use of training cases that indirectly satisfy application conditions that fundamentally distinguishes the invention from prior art methods.

Other classification and regression tree algorithms employ different methods for handling missing data. In such cases, the trees that are constructed typically do not contain tests for missing values. For such algorithms, the same prior art methods for constructing tree branches would be used, except that additional branches must be added to some tree nodes, as per the second half of Method Step 2c, in order to handle missing values using the invention.

Again, the training cases used to construct the models that appear in each tree node would preferably be those that indirectly satisfy the application conditions of the model for those missing values that are mentioned in the application conditions and that are to be treated as missing at random.

Method Step 2d addresses classification and regression rules. As with classification and regression trees, these predictive modeling algorithms also construct a plurality of models. In this case, the plurality is explicitly represented as if-then rules, with the application conditions appearing in the if-parts of the rules, and the subordinate models appearing in the then-parts of the rules. When constructing rules sets, these algorithms not only consider model refinements in which the application conditions of a rule are further restricted (e.g., by adding extra application conditions as with classification and regression trees), but they also consider model refinements whose effects are to relax the application conditions of a role so that the rule is applicable in a wider range of cases (e.g., by eliminating or otherwise generalizing one or more application conditions).

When the latter type of refinement is performed, Method Step 2d specifies that the inputs to the model that appear in the then-part of the resulting rule should preferably be restricted to those inputs that are guaranteed not to have missing values. Other than this preferred restriction, any method for relaxing application conditions can be used in

conjunction with the invention.

As before, the training cases used to construct the models that appear in each rule would preferably be those that indirectly satisfy the application conditions of the rule for those missing values that are mentioned in the application conditions and that are to be treated as missing at random.

Method Step 3 is present in all three of the aforementioned predictive modeling techniques: at some point, model refinement terminates when various stopping conditions are met.

Method Step 4 corresponds to the pruning operation found in classification and regression tree algorithms. Computer methods for implementing post-refinement optimization (e.g., pruning) are well-known to those ordinarily skilled in the art of implementing classification and regression tree algorithms.

Method Step 5 can be implemented by those ordinarily skilled in the art of computer programming. Given a particular combination of data structures for representing a plurality of subordinate models, it should be self-evident how to output a specification of the plurality in such a way that the data structures can be reconstructed when the specification is inputted at a later point in time, perhaps by a separate computer program that applies the plurality to generate predictions.

As previously mentioned, the method steps were phrased in a way that anticipates the possibility of combining aspects of stepwise regression, classification and regression trees, and classification and regression rules in a single algorithm. Hence, Method Steps 2a-d cover each of the various ways of refining a model that are used in the aforementioned predictive modeling methods: adding an input to a model (Step 2a), removing an input from a model (Step 2b), dividing the conditions under which a model is applicable into two or more subcases and building separate models for each subcase (Step 2c), and expanding the conditions under which a model is applicable (Step 2d). Which combination of Method Steps 2a-d are utilized depends on which combination of model refinements are implemented by someone ordinarily skilled in the art of constructing predictive modeling algorithms.

B. Useful, Concrete, Tangible Results

Regarding the Examiner's assertion that "*[as previously explained,] Applicant's claimed invention does not produce a useful concrete, and tangible result, but describes a mathematical algorithm used to construct a predictive model*" (Section 5.2 of the Final Office Action, second to last sentence), Applicant respectfully disagrees.

The Examiner's opinion that the underlying process invention is non-statutory rests on the presumption that Step 5 of the preferred method steps does not produce useful, concrete, and tangible results:

> *Step 5 preferably comprises outputting a specification of the plurality of subordinate models' and their associated application conditions, preferably to a storage device readable by a machine, thereby enabling the plurality to be readily applied to generate predictions.* (See page 23, lines 20-24)

The Examiner's presumption contradicts common practice by those ordinarily skilled in the art of predictive modeling.

It is common practice among data analysts to separate the task of constructing predictive models from the task of applying predictive models to make predictions.

That is, after a predictive model is constructed, it is typically outputted in machine-readable form using a suitable data exchange format so that the model can then be used as input to a computer program that applies the model to make predictions. Such outputting capability is commonly provided by predictive modeling software. Indeed, as Grossman et al. point out (Robert Grossman, Stuart Bailey, Ashok Ramu, Balinder Maihi, Michael Comelison, Philip Hallstrom, and Xiao Qin, "The Management and Mining of Multiple Predictive Models Using the Predictive Modeling Markup Language (PMML)," Armed Forces Communications and Electronics Association (AFCEA) Conference, 1999): "Ever since there has been statistical software, there has been interchange formats for predictive models."

The Predictive Modeling Markup Language (PMML) presented by Grossman et al. is only one example of an interchange format for predictive models. However, PMML is an

important example in that efforts are being made to turn PMML into an open and flexible standard for exchanging predictive models among tools and applications provided by different software vendors (see http://www.dmg.org/).

The existence of predictive model interchange formats render predictive models concrete and tangible. For example, using predictive modeling software and model application software that utilize the same interchange format, one can use predictive modeling software resident on one computer to construct a predictive model and then output that model to a floppy disk. The floppy disk can then be inserted into a separate computer disconnected from the first computer, and model application software resident on the second computer can be used to apply the predictive model to data available to the second computer. The floppy disk is concrete and it tangibly embodies the predictive model.

One of the goals of the PMML standardization effort is, in fact, to enable the above scenario to be played out using predictive modeling software and model application software supplied by two independent vendors, with the PMML encoding of the predictive model preferably transferred by electronic means via a communications network instead of by physical means via a floppy disk.

Thus, the invention clearly provides a useful, concrete, and tangible result.

C. The Examiner's Assertion directed to a "Practical Application"

The Examiner also contends that the output of the invention (e.g., a predictive model) <u>must</u> be limited to a practical application in order for the results (e.g., the predictive model) to be useful.

While it is true that some predictive modeling techniques are designed for specific applications, many are not. A wide variety of predictive modeling techniques are general-purpose in nature and are utilized for specific applications by supplying the software that embodies such techniques with application-specific data. In such cases, no modifications need be made to the techniques nor the software that embodies those techniques. Moreover, the usefulness of the output model is dictated by the usefulness of the input data.

Because general-purpose predictive modeling techniques are general-purpose, they are commonly used as component technologies when building application software. This fact, in conjunction with the increasing prevalence of predictive modeling in business, has motivated

Microsoft Corporation to develop their OLE DB for Data Mining (OLE DB for DM) application programming interface (API). The following excerpt from the Microsoft web document "Introduction to OLE DB for Data Mining" (http://www.microsoft.com/data/oledb/dm.htm) outlines the objectives of this API:

> *Up to now, the data mining industry has been highly fragmented, making it difficult---and costly--for application software vendors and corporate developers to integrate different knowledge-discovery tools. With the help and contributions of more than 40 ISVs in the business intelligence field, Microsoft's OLE DB for DM specification introduces a common interface for data mining that will give developers the opportunity to easily and affordably--embed highly scalable data mining capabilities into their existing applications. Microsoft's objective is to provide the industry standard for data mining so that algorithms from practically any data mining ISV can be easily plugged into a consumer application."*

An important consequence of the OLE DB for DM AP1 is that it effectively commoditizes predictive modeling software by separating such software from the applications that use it. The API thereby enables predictive modeling software provided by one vendor to be substituted for predictive modeling software provided by another vendor without significant changes to the underlying application software.

The commoditization of general-purpose predictive modeling technology implies that the usefulness of such technology is not tied to any specific application. As stated throughout the patent application, the invention is widely applicable and has great general utility. In particular, it can be combined with general-purpose predictive modeling techniques such as stepwise regression, classification and regression trees, and classification and regression roles. Hence, the usefulness of the invention is likewise not tied to any specific application.

Thus, the invention clearly is a statutory product and embraces statutory subject matter clearly worthy of a U.S. Letter Patent.

Additionally, as mentioned in the March 17, 2000 Amendment, Applicant submits that the Office Action references an improper standard with respect to 35 U.S.C. §101.

Applicant again notes the Federal Circuit's decision in AT&T Corp v. Excel Communications, 50 USPQ2d 1447 (Fed. Cir. 1999) (hereafter AT&T v. Excel). This case discusses the current status of 35 U.S.C. §101. Id at 1451. The Federal Circuit states that a process that applies an equation to a new and useful end is at the very least not barred by the

threshold by §101. Furthermore, a claimed processing system for implementing a financial management system (as in State Street) constituted a practical application of a mathematical algorithm by producing "a useful, concrete and tangible result." Id at 1451. Furthermore, in discussing State Street, the Federal Circuit held that there was patentable subject matter because the system takes data representing discrete dollar amounts through a series of mathematical calculations to determine a final share price, which was considered a useful, concrete and tangible result. See page 1452.

The Court also suggests that the notion of a physical transformation (as alluded to in the present Office Action) is but one example of how a mathematical algorithm may bring about a useful application. Therefore, Applicant submits that the Office Action makes an improper rejection under 35 U.S.C. §101. Patentability under 35 U.S.C. §101 requires a determination of whether a useful, concrete and tangible result is accomplished by the claimed features. As discussed above, Applicant submits that it has developed a useful, concrete and tangible result from the claimed features, the utility being clearly described in the application as discussed above.

As mentioned above, the present invention relates to method for constructing a predictive model that can be used for making predictions (e.g., reliable ones on which a decision may be based) even when the values of some or all inputs are missing or are otherwise unknown. See pages 1, lines 9-12; page 2, lines 9-14, etc.). Accordingly, the present invention provides a predictive model which is superior to the prior art and which can make superior predictions from those of the prior art models, even when some or all data values are unknown or missing. This obviously relates to making real-world predictions for real-world problems and decisions. Indeed, on pages 1-2 a real world, exemplary application of direct-mail targeted-marketing purposes in industries that sell directly to consumers. Clearly, such an application is a useful, concrete and tangible result especially in the direct-mail industry. By the same token, Applicant submits that the claims need not specifically recite such an exemplary application, thereby limiting the claims only to such an exemplary application. Indeed, to do so would be improper (and foolhardy) for Applicant.

Furthermore, the present application discusses the problems of the prior art and how the present invention overcomes such problems (and in some cases can be used with the conventional methods of model generation) (e.g., see pages 2-7 and 23-26 of the present

application).

Pages 10-22 describe features of the present invention and the respective features that obtain the objectives of the present application. The detailed description discusses the use of the invention in the form of a special apparatus or computer program executed in a generally used computer (e.g., see page 19, lines 22-24). One skilled in the art would clearly understand that this stored data may be read from the computer such as on a display unit, an output unit such as a printer, etc. Clearly, the generation of such predictive models provides a useful, concrete and tangible result for at least the direct-marketing industry (e.g., see page 1, line 29 to page 2, line 4). Indeed, such predicative models allow predictions to be generated with increased reliability despite their being missing values (e.g., possibly missing demographic, credit or other data inputs), and allows a greater return on marketing investments in this particular application.

The claims describe the program storage device for storing the method for constructing predictive models that can be used for making such predictions despite the presence of missing data values. The generation of a predictive model is patentable subject matter if it is a practical application that produces a useful, concrete and tangible result. See AT&T v. Excel. It is clear that this invention as a whole is applied in a useful manner, as described above (i.e., it is useful to generate a predictive model which will generate predictions having increased reliability and upon which marketing and financial decisions may be made). The independent claims set forth very detailed steps of how to arrive at this result so as to avoid problems of the prior art.

There is no requirement that the claims set forth any post-computational activity as asserted in the Office Action. Rather, as discussed on page 1452 of AT&T v. Excel, a physical transformation is merely one example of how a mathematical algorithm may bring about a useful application. In this application, the construction of the subordinate models (e.g., as defined in (2) in independent claim 1) and specifically the testing of inputs, and treatment of known data values and missing values in the claim results in the construction of a model which has increasingly reliable predictions as compared to the prior art and which avoids the problems of the prior art. By avoiding these problems, the present application provides a useful, concrete and tangible result and therefore the requirements of 35 U.S.C. §101 are met.

Additionally, Applicant again points out that claim 1 recites, <u>inter alia</u>, "*outputting a specification of at least one of said subordinate models thus generated and making a prediction based on said at least one of said subordinate models thus-generated*". This clearly defines a "post-computation/mathematical operation processing" and clearly the subject matter of independent claim 1 (and claims 2-20 which depend from claim 1) is statutory and allowable over the prior art of record.

In view of all of the foregoing, reconsideration and withdrawal of the rejection is respectfully requested.

III. CONCLUSION

In view of the foregoing, Applicant submits that claims 1-20, all the claims presently pending in the application, are patentably distinct over the prior art of record and are in condition for allowance. The Examiner is respectfully requested to pass the above application to issue at the earliest possible time.

Should the Examiner find the application to be other than in condition for allowance, the Examiner is requested to contact the undersigned at the local telephone number listed below to discuss any other changes deemed necessary in a <u>telephonic or personal interview</u>.

The Commissioner is hereby authorized to charge any deficiency in fees or to credit any overpayment in fees to Attorney's Deposit Account No. 50-0481.

Respectfully Submitted,

Sean M. McGinn

Date: 8/2/00

Sean M. McGinn, Esq.
Reg. No. 34,386

McGinn & Gibb, P.C.
1701 Clarendon Boulevard, Suite 100
Arlington, Virginia 22209
(703) 294-6699
Customer No. 21254